REMARKS/ARGUMENTS

Claims 2-9, 19-21, 23-24, 37-44, 51-54, and 62 are currently pending in the present patent application, claims 1, 11-15, 17-18, 22, and 46-50 are amended, claims 10, 16, and 45 are cancelled, and claims 25-36 and 55-61 are withdrawn. In view of at least the following, all currently pending claims are in condition for allowance, and, therefore, the Applicants' attorney requests that the Examiner withdraw all outstanding rejections. ***However, if after considering this response the Examiner does not allow all of the claims, the Applicants' attorney requests that the Examiner contact him to schedule a telephone interview to further the prosecution of this application.***

Rejection Of Claims 1-4, 6, 8, 10-13, 15, 18, 37-42, And 45-49 Under 35 U.S.C. § 102(e) As Being Anticipated By U.S. 6,405,266 To Bass

Claim 1

Independent claim 1 as previously pending recites first and second parallel buffers respectively associated with first and second data-processing units, and a processor operable to publish data, to load at least a portion of the published data into the first buffer under the control of a first data-transfer object, to load at least the same portion of the published data into the second buffer under the control of a second data-transfer object, and to retrieve at least the portion of the published data from the first and second buffers under the control of third and fourth data-transfer objects, respectively.

For example, referring, *e.g.*, to FIGS. 3-5 and paragraphs [67] – [72] and [83] of the patent application, in an embodiment, a processor 42 is operable, under the control of an application thread 100_3, to publish data, and is operable, under the control of data-transfer object 86_{3a}, and to load at least a portion of the published data into a first buffer 106_3, which is associated with a first data-processing unit (*e.g.*, a first hardwired pipeline 74) within the pipeline accelerator 44. The processor is further operable, under the control of data-transfer object 86_{5a}, to load at least the same portion of the published

data into a second buffer 106_5, which is associated with a second data-processing unit (*e.g.*, a second hardwired pipeline 74) within the pipeline accelerator and which is parallel to the first buffer 106_3, and is operable retrieve at least the portion of the published data from the first and second buffers 106_3 and 106_5 under the control of third and fourth data-transfer objects 86_{3b} and 86_{5b}, respectively.

Referring to FIG. 1 of Bass and to section (8) of the office action, the examiner's reading of claim 1 onto Bass seems to be as follows. A first buffer is located between a publishing object 1 (the first data-transfer object of claim 1) and a subscribing object 2 (the examiner does not point out the exact location of this buffer), and the object 1 publishes data and loads the data into this first buffer. A second buffer is located between the object 1 and a subscribing object 4 (the examiner does not point out the exact location of this buffer, although per below it must be after the message broker 8), and the message broker 8 (the second data-transfer object of claim 1) loads the published data from the object 1 into this second buffer. Data is retrieved from the first buffer by the object 2 (the third data-transfer object of claim 1), and from the second buffer by the object 4 and/or the message broker 113 (the fourth data-transfer object of claim 1).

But the examiner's reading of claim 1 onto Bass, and thus the examiner's rejection of claim 1, fails, because despite the examiner's statement that the first and second buffers of claim 1 are inherent in Bass, the applicants' attorney can find no mention of buffers in Bass. And even if buffers are inherent in Bass, the applicants' attorney can find nothing in Bass from which one may determine the locations of such buffers, let alone determine that Bass includes buffers that read on the buffers recited in claim 1.

And the examiner's rejection of claim 1 over Bass also fails because it appears to be impossible for Bass to read on both the first and second data-processing units and the processor as recited in claim 1. Claim 1 recites that the first and second buffers are respectively associated with first and second data-processing units, and that a processor executes all of the four data-transfer objects. To meet this first limitation, the examiner must interpret Bass to teach that process A and process B run on different data-processing units, *e.g.*, different processors, because the examiner's first buffer

corresponds to Bass's object 2 (executed by process A), and the examiner's second buffer corresponds to Bass's object 4 (executed by process B). But to meet the second limitation, the examiner must interpret Bass to teach that process A and process B run on the same processor such that object 2 and object 4 (and/or message broker 113) are executed by the same processor. Consequently, because these two interpretations of Bass are mutually exclusive, Bass does not and cannot anticipate claim 1.

Claim 2

Dependent claim 2 as previously pending recites that the first and third data-transfer objects respectively comprise first and second instances of first object code, and the second and fourth data-transfer objects respectively comprise first and second instances of second object code.

In contrast, Bass does not disclose multiple objects being instantiations of the same object code. Contrary to the examiner's position in section 9 of the office action, that two objects perform a similar function (message communicating) does not require that they are instantiations of the same object code.

Furthermore, claim 2 is patentable at least by virtue of its dependency from claim 1.

Claim 3

Claim 3 is patentable at least by virtue of its dependency from claim 1.

Claim 4

Dependent claim 4 as previously pending recites that the processor is further operable to execute a thread of the application and to publish the data under the control of the thread.

In contrast to the examiner's position in section 11 of the office action, Bass appears to disclose only that subscribing objects, not publishing objects, are executed as part of a thread.

Furthermore, claim 4 is patentable at least by virtue of its dependency from claim 1.

Claim 6

Dependent claim 6 as previously pending recites wherein the processor is operable to drive the data retrieved from one of the first and second buffers onto a bus.

In contrast, as discussed above in support of the patentability of claim 1, Bass does not disclose the first and second buffers.

Furthermore claim 6 is patentable at least by virtue of its dependency from claim 1.

Claim 8

Dependent claim 8 as previously pending recites that the processor is further operable to generate a message that includes the data retrieved from one of the first and second buffers.

In contrast, as discussed above in support of the patentability of claim 1, Bass does not disclose the first and second buffers.

Furthermore, claim 8 is patentable at least by virtue of its dependency from claim 1.

Claim 10

Claim 10 is cancelled.

Claim 11

Claim 11, which as been amended to be in independent form and to include all of the limitations of claim 10, recites that the first and second data-transfer objects respectively comprise first and second instances of the same object code.

In contrast, Bass does not disclose multiple objects being instantiations of the same object code. Contrary to the examiner's position in section 15 of the office action, that two objects perform a similar function (message communicating) does not require that they are instantiations of the same object code.

Claim 12

Claim 12 as amended is patentable at least by virtue of its dependency from claim 11.

Claim 13

Dependent claim 13 as amended recites generating the data under the control of a thread.

In contrast to the examiner's position in section 17 of the office action, Bass appears to disclose only that subscribing objects, not publishing (*i.e.*, data-generating) objects, are executed as part of a thread.

Claim 15

Claim 15 as amended is patentable at least by virtue of its dependency from claim 11.

Claim 18

Claim 18 as amended is patentable at least by virtue of its dependency from claim 11.

Claim 37

Claim 37 as previously pending recites loading published data into a buffer, each location within the buffer corresponding to an address of a destination of the data.

In contrast to the examiner's position in section 20 of the office action, Bass discloses no buffer for receiving published data, where each location of the buffer corresponds to an address of a destination of the data. Col. 2, lines 18-25 of Bass state only that a subscriber A's request for information from a publisher B is queued. Bass does not disclose that the information from B, once published, is queued. Furthermore, Bass does not state that the queue corresponds to A, which would be the destination of the information requested from B once B publishes this information; Bass states only that a PUB/SUB broker forwards A's request from the queue to B. And there is nothing in Bass that would indicate that the queue is dedicated to (*i.e.*, that each location within the queue corresponds to an address of) the process A. That is, there appears to be nothing in Bass to indicate that the queue cannot hold requests from processes other than A.

Claim 38

Dependent claim 38 as previously pending recites publishing the data with a thread of the application.

In contrast to the examiner's position in section 21 of the office action, Bass appears to disclose only that subscribing objects, not publishing objects, are executed as part of a thread.

Claim 38 is also patentable at least by virtue of its dependency from claim 37.

Claim 39

Dependent claim 39 as previously pending recites generating a queue value that corresponds to the presence of the published data in the buffer.

In contrast to the examiner's position in section 22 of the office action, it appears that Bass does not disclose generating a queue value that corresponds to the presence of published data in a buffer. As discussed above in support of the patentability of claim 37, Bass does not disclose the queuing of published data. And the examiner's position that it is inherent in Bass that "some signal /value must be generated which indicates data is received in a buffer and that it must be retrieved" appears to be incorrect because other techniques (*e.g.*, polling) exist to determine the presence of data in a buffer.

Claim 39 is also patentable at least by virtue of its dependency from claim 37.

Claim 40

Claim 40 as previously pending is patentable at least by virtue of its dependency from claim 37.

Claim 41

Dependent claim 41 as previously pending recites loading the retrieved data into a second buffer.

In contrast to the examiner's position in section 24 of the office action, Bass discloses nothing about buffering published data as discussed above in support of the patentability of claim 37; therefore, the examiner's statement that one of Bass's brokers

would load data from one buffer to another is mere speculation and is not taught or suggested by Bass.

Claim 41 is also patentable at least by virtue of its dependency from claim 37.

Claim 42

Claim 42 is patentable at least by virtue of its dependency from claim 37.

Claim 45

Claim 45 is cancelled.

Claim 46

Claim 46 as amended is patentable at least by virtue of its dependency from claim 47.

Claim 47

Claim 47, which has been amended to be in independent form and to include all of the limitations of cancelled claim 45, recites generating a queue value that corresponds to the presence of data in a buffer.

In contrast to the examiner's position in section 28 of the office action, it appears that Bass does not disclose generating a queue value that corresponds to the presence of data in a buffer. The examiner's position that it is inherent in Bass that "some signal/value must be generated which indicates data is received in a buffer and that it must be retrieved/unloaded" appears to be incorrect because other techniques (*e.g.*, polling) that do not involve generating a signal value to indicate the presence of data in a buffer.

Claim 48

Claim 48 as amended is patentable at least by virtue of its dependency from claim 47.

Claim 49

Claim 49 as amended is patentable at least by virtue of its dependency from claim 47.

Rejection Of Claim 7 Under 35 U.S.C. § 103(a) As Being Unpatentable Over Bass

Dependent claim 7 as previously pending recites that the processor is operable to provide the data retrieved from one of the first and second buffers to a third buffer.

In contrast to the examiner's position in section 33 of the office action, Bass does not disclose or suggest providing published data retrieved from one of first and second buffers to a third buffer. The examiner states "one of ordinary skill in the art would have recognized that there may be more than just two subscribing objects." But even if the examiner is correct, recognizing that there may be more that two subscribing objects does not mean that the buffer scheme recited in claim 7 would have been obvious, particularly considering that Bass mentions nothing about buffering published data (see col. 2, lines 19-26, which appears to talk only about queuing subscriber requests).

Furthermore, Claim 7 is patentable at least by virtue of its dependency from claim 1.

Rejection Of Claims 5, 9, 14, 17, 43-44, and 50 Under 35 U.S.C. § 103(a) As Being Unpatentable Over Bass In View Of The Examiner's Taking Of Official Notice

The applicants' attorney requests that the examiner cite one or more references to show the facts of which he is taking official notice, and to clarify the facts of which he is taking official notice as requested below.

Claim 5

Dependent claim 5 as previously pending recites storing a queue value under the control of a queue object, the queue value reflecting the loading of published data into a buffer.

In contrast to the examiner's statement in section 35 of the office action, it is not inherent in Bass to "store a queue value, the queue value reflecting the loading of the published data into the first buffer." As discussed above in support of the patentability of claim 37, Bass does not disclose the queuing or buffering of published data. And because Bass does not even mention buffering, and discusses queuing only in conjunction with subscriber requests, Bass does not suggest the queuing or buffering of published data.

Furthermore, it is unclear of what facts or conclusions the examiner has taken Official Notice regarding claim 5. Therefore, the examiner is requested to clarify the facts of which he is taking Official Notice.

In addition, claim 5 is patentable by virtue of its dependency on claim 1.

Claim 9

Dependent claim 9 as previously pending recites that first and third data-transfer objects respectively comprise first and second instances of first object code, second and fourth data-transfer objects respectively comprise first and second instances of second object code, and the processor is operable to execute an object factory and to generate the first object code and the second object code under the control of the object factory.

In contrast to the examiner's positions in sections 36(a) and 36(b) of the office action, Bass does not teach first and third data-transfer objects respectively comprise first and second instances of first object code, and second and fourth data-transfer objects respectively comprise first and second instances of second object code. That two objects comprise the same type of code (*e.g.*, message-communicating code) does not imply that they are instantiations of the same object code. And Bass does not suggest two objects being instantiations of the same object code because what the examiner calls "objects" in Bass perform different functions, and so Bass instead suggests objects that are instantiations of different object code.

And in contrast to the examiner's position in section 36(c) of the office action, it would not have been obvious to modify Bass such that a processor is operable to execute an object factory and to generate first object code and second object code

under the control of an object factory. Generally, an object factory generates object code in response to information contained in a registry. But Bass includes no teaching or suggestion of such a registry; consequently, regarding the objects 1-4 in FIG. 1, Bass includes no teaching or suggestion that these objects include code generated under the control of an object factory. And regarding the brokers 103, 108, and 113 of FIG. 1, Bass actually teaches away from an object factory generating them. Referring to FIG. 1 and to col. 6, lines 6-23, a human administrator manually configures the brokers 103, 108, and 113, and thus the code that forms these brokers, via a graphical user interface (GUI) 116.

Furthermore, it is unclear of what facts the examiner has taken Official Notice to support his rejection of claim 9. Therefore, the examiner is requested to clarify the facts of which he is taking Official Notice.

In addition, claim 9 is patentable by virtue of its dependency on claim 1.

Claim 14

Dependent claim 14 as amended recites storing a queue value under the control of a queue object, the queue value reflecting the loading of data into a buffer, the data having been generated under the control of an application.

In contrast to the examiner's statement in section 37 of the office action, it is not inherent or suggested in Bass to "store a queue value, the queue value reflecting the loading of the retrieved data into the first buffer." As discussed above in support of the patentability of claim 37, Bass appears to disclose only queuing of subscriber requests for published data, not the queuing or buffering of data generated under the control of an application.

Furthermore, it is unclear of what facts or conclusions the examiner has taken Official Notice regarding claim 14. Therefore, the examiner is requested to clarify the facts of which he is taking Official Notice.

In addition, claim 14 is patentable by virtue of its dependency on claim 11.

Claim 17

Dependent claim 17 as amended recites that the first and second data-transfer objects respectively comprise first and second instances of the same object code, and the processor is operable to execute an object factory and to generate the object code under the control of the object factory.

In contrast to the examiner's position in section 38(a) of the office action, Bass does not teach first and second data-transfer objects respectively comprise first and second instances of the same object code. That two objects comprise the same type of code (*e.g.*, message-communicating code) does not imply or suggest that they are instantiations of the same object code.

And in contrast to the examiner's position in section 38(b) of the office action, it would not have been obvious to modify Bass such that a processor is operable to execute an object factory and to generate the object code under the control of an object factory. Generally, an object factory generates object code in response to information contained in a registry. But Bass includes no teaching or suggestion of such a registry; consequently, regarding the objects 1-4 in FIG. 1, Bass includes no teaching or suggestion that these objects include code generated under the control of an object factory. And regarding the brokers 103, 108, and 113 of FIG. 1, Bass actually teaches away from an object factory generating them. Referring to FIG. 1 and to col. 6, lines 6-23, a human administrator manually configures the brokers 103, 108, and 113, and thus the code that forms these brokers, via a graphical user interface (GUI) 116.

Furthermore, it is unclear of what facts the examiner has taken Official Notice to support his rejection of claim 17. Therefore, the examiner is requested to clarify the facts of which he is taking Official Notice.

In addition, claim 17 is patentable by virtue of its dependency on claim 11.

Claim 43

Dependent claim 43 as previously pending recites generating data-transfer object code with an object factory, generating a first data-transfer object as a first instance of the object code, and generating a second data-transfer object as a second instance of the object code.

In contrast to the examiner's position in section 39(a) of the office action, Bass does not teach generating first and second data-transfer objects as respective first and second instances of the same object code. That two objects comprise the same type of code (*e.g.*, message-communicating code) does not imply or suggest that they are instantiations of the same object code.

And in contrast to the examiner's position in section 39(b) of the office action, it would not have been obvious to modify Bass to generate object code with an object factory. Generally, an object factory generates object code in response to information contained in a registry. But Bass includes no teaching or suggestion of such a registry; consequently, regarding the objects 1-4 in FIG. 1, Bass includes no teaching or suggestion that these objects include code generated under the control of an object factory. And regarding the brokers 103, 108, and 113 of FIG. 1, Bass actually teaches away from an object factory generating them. Referring to FIG. 1 and to col. 6, lines 6-23, a human administrator manually configures the brokers 103, 108, and 113, and thus the code that forms these brokers, via a graphical user interface (GUI) 116.

Furthermore, it is unclear of what facts the examiner has taken Official Notice to support his rejection of claim 43. Therefore, the examiner is requested to clarify the facts of which he is taking Official Notice.

In addition, claim 43 is patentable by virtue of its dependency on claim 37.

Claim 44

Dependent claim 44 as previously pending recites receiving a message and processing data in the message with a hardwired pipeline accelerator.

The applicants' attorney objects to the examiner taking Official Notice that hardwired-pipeline processors and their advantages are well-known and accepted in the art; if hardwired-pipeline processors and their advantages are well known, than it should be relatively easy for the examiner to provide a prior-art reference that shows this.

Furthermore, even if the examiner is correct that hardwired-pipeline processors and their advantages are well-known and accepted in the art, this is immaterial, because claim 44 recites "a hardwired pipeline accelerator," not a hardwired-pipeline processor. Therefore, if the examiner intends to maintain this rejection, the applicants'

attorney requests him to provide at least one prior-art reference that shows a hardwired-pipeline accelerator that receives a message and that processes data in the message.

In addition, the examiner is requested to specifically identify what he calls "Bass's receiving processor."

Moreover, claim 44 is patentable by virtue of its dependency on claim 37.

Claim 50

Dependent claim 50 as amended recites generating a message header and a message with a hardwired pipeline accelerator.

The applicants' attorney objects to the examiner taking Official Notice that hardwired-pipeline processors and their advantages are well-known and accepted in the art; if hardwired-pipeline processors and their advantages are well known, than it should be relatively easy for the examiner to provide a prior-art reference that shows this.

Furthermore, even if the examiner is correct that hardwired-pipeline processors and their advantages are well-known and accepted in the art, this is immaterial, because claim 44 recites "a hardwired pipeline accelerator," not a hardwired-pipeline processor. Therefore, if the examiner intends to maintain this rejection, the applicants' attorney requests him to provide at least one prior-art reference that shows a hardwired-pipeline accelerator that generates a message header and a message.

Moreover, claim 50 is patentable by virtue of its dependency on claim 47.

Rejection Of Claims 10-15, 17-18, And 45-50 Under 35 U.S.C. § 103(a) As Being Unpatenable Over U.S. Patent 4,703,475 To Dretzka In View Of The Examiner's Taking Of Official Notice

Claim 10

Claim 10 has been cancelled.

Claim 11

Claim 11 has been amended to be in independent form to include all of the limitations of cancelled claim 1. Therefore, the applicants' attorney addresses the examiner's rejections of claims 10 and 11 here.

Furthermore, the applicants' attorney addresses only the portions of the examiner's rejections that are new relative to the office action mailed 08 June 2010; the applicants' attorney already addressed the old portions of the examiner's rejections in the response filed 08 October 2010.

Claim 11 as amended recites a processor operable to execute first and second data-transfer objects and an application, to generate data under the control of the application without generating an address of a destination of the data, to retrieve the generated data from the application and to load the retrieved data into the buffer under the control of the first data-transfer object, to unload the data from the buffer under the control of the second data-transfer object, to process the unloaded data under the control of the application without receiving the address of the destination of the data, and wherein the first and second data-transfer objects respectively comprise first and second instances of the same object code.

The examiner's first position in sections 43(b2), 50(b1), and 68 of the office action appears to be that because Dretzka's data is separate from the address, that Dretzka discloses generating the data without generating an address of a destination of the data.

Even if the examiner is correct that Dretzka's data is separate from the address, this is immaterial for at least two reasons. First, claim 11 does not recite that the data is separate from the address, but recites generating "data under the control of the

application without generating an address of a destination of the data." Second, even if Dretzka's application generates the data separate from the address of the data's destination, as explained in the response filed 08 October 2010, Dretzka's application cannot generate data without generating the data's destination address because without the address, Dretzka's circuitry would have no way of determining where to send the data.

The examiner's second position in sections 43(b4), 50(b4), and 68 of the office action appears to be that because Dretzka's data is processed separately from the address, that Dretzka discloses processing the data without receiving an address of a destination of the data.

Even if the examiner is correct that Dretzka's data is processed separately from the address, this is immaterial for at least two reasons. First, claim 11 does not recite that the data is processed separately from the address, but recites processing the unloaded data under the control of the application without receiving the address of the destination of the data." Second, even if Dretzka's application may process the data separate from the address of the data's destination, as explained in the response filed 08 October 2010, Dretzka's application cannot process data without first receiving the data's destination address because without the address, Dretzka's processing circuitry would have no way of determining whether the data was intended for it or for another destination.

The examiner's third position in sections 44 and 51 of the office action seems to be that because two objects perform the same type of function, and thus may have the same type of object code, it is inherent or obvious that the objects are instantiations of the same object code. But this is incorrect, because objects may comprise different object code and still perform the same type of function. And nowhere does Dretzka disclose or suggest objects that are instantiations of the same object code; in fact, even the examiner admits that Dretzka does not disclose objects.

Furthermore, it is unclear of what facts the examiner has taken Official Notice to support his rejection of claims 10 and 11. Therefore, the examiner is requested to clarify the facts of which he is taking Official Notice.

Claim 12

In sections 45 and 52 of the office action, it is unclear of what facts the examiner has taken Official Notice regarding claim 12. Therefore, the examiner is requested to clarify the facts of which he is taking Official Notice.

In addition, claim 12 is patentable by virtue of its dependency on claim 11.

Claim 13

Regarding sections 46 and 53 of the office action, the applicants' attorney objects to the examiner taking Official Notice that it is "known to divide up a program into threads in order to increase efficiency by reducing stall time;" if this is so well known, than it should be relatively easy for the examiner to provide a prior-art reference that shows this.

Furthermore, Dretzka does not disclose or suggest patentable for the reasons similar to those submitted in the response mailed 08 October 2010.

In addition, claim 13 is patentable by virtue of its dependency on claim 11.

Claims 14-15

In sections 47-48 and 54-55 of the office action, it is unclear of what facts the examiner has taken Official Notice regarding claims 14-15. Therefore, the examiner is requested to clarify the facts of which he is taking Official Notice.

In addition, claims 14-15 are patentable by virtue of their dependencies on claim 11.

Claim 17

Dependent claim 17 as amended recites the first and second data-transfer objects respectively comprise first and second instances of the same object code, and the processor is operable to execute an object factory and to generate the object code under the control of the object factory.

In contrast to the examiner's positions in sections 49(a) and 56(a) of the office action, Dretzka does not teach first and second data-transfer objects respectively comprising first and second instances of the same object code. That two objects

comprise the same type of code (*e.g.*, retrieve-and-store) does not imply that they are instantiations of the same object code.

And in contrast to the examiner's positions in sections 49(b) and 56(b) of the office action, it would not have been obvious to modify Dretzka such that a processor is operable to execute an object factory and to generate the object code under the control of an object factory. Generally, an object factory generates object code in response to information contained in a registry. But Dretzka includes no teaching or suggestion of such a registry. Furthermore, as the examiner points out, Dretzka does not even mention objects. Therefore, Dretzka does not make obvious an object factory when he does not event mention objects, object-code-building data, or a registry where such building data may be stored. Moreover, in section 56(b) of the office action, the examiner states that "when the program calls for data to be transmitted," object code for data-transfer objects "will be generated [by Dretzka's circuitry] and invoked so that data may be transmitted." But Dretzka nowhere discloses or suggests such dynamic generation of objects. In fact, Dretzka inherently teaches away from such dynamic generation of objects because it appears that to generate objects each time that a program calls for the transmission of data would slow down the system to a point where it would no longer be useful for its intended purpose.

Furthermore, it is unclear of what facts the examiner has taken Official Notice to support his rejection of claim 17. Therefore, the examiner is requested to clarify the facts of which he is taking Official Notice.

In addition, claim 17 is patentable by virtue of its dependency on claim 11.

Claim 18

In section 57, it is unclear of what facts the examiner has taken Official Notice regarding claim 18. Therefore, the examiner is requested to clarify the facts of which he is taking Official Notice.

In addition, claim 18 is patentable by virtue of its dependency on claim 11.

Claim 47

Claim 47 has been amended to be in independent form and to include all of the

limitations of cancelled claim 45. Therefore, the applicants' attorney addresses the examiner's rejections of claims 45 and 47 here.

Furthermore, the applicants' attorney addresses only the portions of the examiner's rejections that are new relative to the office action mailed 08 June 2010; the applicants' attorney already addressed the old portions of the examiner's rejections in the response filed 08 October 2010.

Claim 47 as amended recites receiving a message that includes data and that includes a message header that indicates a destination address of the data, loading into a buffer, with a first data-transfer object, the received data, unloading the data from the buffer with a second data-transfer object, and processing the unloaded data with a software application without the software application receiving the destination address of the data.

In contrast, Dretzka does not disclose or suggest loading and retrieving data from a buffer with respective data-transfer objects, and does not disclose processing the unloaded data with a software application that does not receive the destination address of the data. The examiner has failed to address the latter limitation in either section 58(d) or 69; consequently, the applicants' attorney requests the examiner to either address this limitation or allow claim 47.

Furthermore, Dretzka does not disclose or suggest receiving a message.

The examiner seems to think that a message can be any grouping of data and a header.

But as discussed below, the examiner is incorrect.

A message is a complete data structure that is in a format compatible with the interface of a software object that receives or transfers the message (see, *e.g.*, Wikipedia and the reference "Design Patterns: Elements of Reusable Object-Oriented Software," selected passages of which are cited in the accompanying Information Disclosure Statement).

In contrast, Dretzka fails to disclose or suggest a complete data structure that is in a format compatible with a software-object interface. Dretzka teaches data groups that are purposely divided into packets for transmission over links 40; consequently, these divided data groups are not complete data structures. And Dretzka does not even

discuss software objects or interfaces thereto, let alone a data structure that is compatible with a software-object interface.

In sections 58(e) and 69, the examiner states that because it would have been obvious to implement, *e.g.*, Dretzka's loaders and unloaders, as objects, it would have been obvious to modify Dretzka's messages to "be compatible with an object interface."

But the examiner is incorrect, at least because to modify Dretzka's messages in such a manner would require a significant redesign of Dretzka's system, and the examiner has not shown that such a redesign would provide advantages significant enough to motivate one to undertake such a redesign.

Furthermore, in sections 58, 60, and 69, it is unclear of what facts the examiner has taken Official Notice regarding claims 45 and 47. Therefore, the examiner is requested to clarify the facts of which he is taking Official Notice.

Claim 46

Dependent claim 46 as amended recites processing the unloaded data with a thread of the software application.

Regarding section 59 of the office action, the applicants' attorney objects to the examiner taking Official Notice that it is "known to divide up a program into threads in order to increase efficiency by reducing stall time;" if this is so well known, than it should be relatively easy for the examiner to provide a prior-art reference that shows this.

Furthermore, claim 46 is patentable by virtue of its dependency on claim 47.

Claims 48-49

In sections 61-62, it is unclear of what facts the examiner has taken Official Notice regarding claims 48-49. Therefore, the examiner is requested to clarify the facts of which he is taking Official Notice.

In addition, claims 48-49 are patentable by virtue of their dependencies on claim 47.

Claim 50

Dependent claim 50 as amended recites generating a message header and a message with a hardwired pipeline accelerator.

The applicants' attorney objects to the examiner taking Official Notice that hardwired-pipeline processors and their advantages are well-known and accepted in the art; if hardwired-pipeline processors and their advantages are well known, than it should be relatively easy for the examiner to provide a prior-art reference that shows this.

Furthermore, even if the examiner is correct that hardwired-pipeline processors and their advantages are well-known and accepted in the art, this is immaterial, because claim 44 recites "a hardwired pipeline accelerator," not a hardwired-pipeline processor. Therefore, if the examiner intends to maintain this rejection, the applicants' attorney requests him to provide at least one prior-art reference that shows a hardwired-pipeline accelerator that generates a message header and a message.

Moreover, claim 50 is patentable by virtue of its dependency on claim 47.

Allowable Subject Matter

The applicants' attorney thanks the examiner for allowing claims 19-24, 51-54, and 62 in section 64 of the office action.

The applicants' attorney does not necessarily agree with the examiner's interpretations of "object" and "publish" in sections 65 and 66 of the office action, and that these and the other claim limitations should be interpreted as a court of competent jurisdiction would interpret them.

Conclusion

The absence of additional patentability arguments should not be construed as either a disclaimer of such arguments or that such arguments are not believed to be meritorious. In light of at least the reasons discussed herein, in addition to the allowed claims 19-24, 51-54, and 62, claims 2-9, and 37-44 as previously pending and claims 1, 11-15, 17-18, 22, and 46-50 as amended are in condition for allowance. Favorable consideration and a Notice of Allowance are respectfully requested. Should the Examiner have any further questions about the application, Applicant respectfully requests the Examiner to contact the undersigned attorney at (425) 455-5575 to resolve the matter.

In the event additional fees are due as a result of this amendment, you are hereby authorized to charge such payment to Deposit Account No. 07-1897.

DATED this 28th day of February 2011.

Respectfully submitted,

GRAYBEAL JACKSON LLP

/Bryan A. Santarelli/

Bryan A. Santarelli
Registration No. 37,560

Customer No. 00996

Graybeal Jackson LLP
400 - 108th Avenue NE, Suite 700
Bellevue, Washington 98004-5565
Telephone: 425.455.5575
Facsimile: 425.455.1046